痕跡を用いた侵入検出手法への正規手続きデータベースを利用した侵入判定の適用

原田慎介[†] 浅香緑[††]

従来の侵入検出技術ではシステムへ高い負荷をかけずに未知の侵入手法を検出することはできない．そこで我々は痕跡という概念に基づいた侵入検出手法を提案した．この手法は，痕跡という多くの侵入行為に共通する侵入行為の目的の特徴を定義して重点的に監視することで，侵入が検出可能になるというものである．これにより侵入解析に必要な情報量を削減できた．しかしこの提案では，未知の手法による侵入を十分に検出することはできなかった．そこで本論文では，この痕跡を用いた侵入検出手法へ，正規手続きデータベースを用いた侵入判定手法を適用することを提案している．この手法は痕跡があがった際に，それを引き起こしたプロセスが正規に認められた行為かということを，正規手続きデータベースと照らし合わせて判定するというものである．これにより侵入手法に関する知識がなくても侵入検出が可能になる．本論文では正規手続きデータベースを利用した侵入判定手法を提案するとともに，その評価結果を報告する．

Applying Legitimate Access Database for MLSI-based Intrusion Detection

Shinsuke Harada[†] and Midori Asaka[††]

Currently it is impossible to detect intruders who use unknown exploit without overhead. So we proposed a new intrusion detection method based on MLSI (Marks Left by Suspected Intruders). It is possible to detect intruder without any knowledge about each intrusion technique by auditing common features that must be seen when someone intrudes on computer by any intrusion method. Hence if we only audit MLSIs, we can reduce information that should be audited. And now, we propose a new intrusion judgement method by using "legitimate access data base". This method jugdes legitimacy (or not) of a process that caused MLSIs from the legitimate access data base that describes which access is allowd. It is possible to judge MLSIs without kowledge about each intrusion technique.

1. はじめに

現在，インターネットに接続されたホストへの侵入が数多く報告されている[1),2)]．侵入を防ぐ技術として，ファイアウォールを用いたアクセス制御技術，パスワードによる認証技術があるが，以下のような理由により，これらの防御技術だけでホストを侵入から完全に守ることはできない[3)]．

- システムが巨大化するにともない，プログラムからバグを完全になくすことがより困難となり，また，運用上の設定ミスによる弱点が生じる可能性も増える．そうした弱点やバグからの侵入は上記の技術では防ぐことができない．
- パスワードによる認証は，パスワードの紛失，盗難やパスワード破りという脅威に対して無力である．
- 防御技術によって堅固なシステムを構築しても，内部にアクセス権限を持つユーザによる犯行は防ぐことができない．

そこで，近年これらの防御技術だけでなく，侵入を検出するためのシステム（IDS: Intrusion Detection System）の研究が行われている．

これまでの研究において侵入検出手法は，通常のユーザアクティビティや通常のシステムの状態のプロファイルを作成して，プロファイルから大きくはずれた行為を侵入と定義する AID（Anomaly Intrusion Detection）と，既知の侵入手法をルールベースとして持っていて，同じパターンのユーザの行動を侵入と

† 慶應義塾大学理工学研究科
Faculty of Engineering, Keio University
†† 情報処理振興事業協会
Information-technology Promotion Agency, Japan

して検出する MID（Misuse Intrusion Detection）の2種類に分類される[4]．これらの手法では，システムに負荷をかけずに未知の侵入手法を検出することができない．そこで我々は，システムに負荷をかけずに未知の手法を検出する IDS の構築を目的とした．

我々はこれまでに，痕跡という概念に基づいた侵入検出手法を提案した[5]．この手法は，多くの侵入行為に共通する侵入行為の目的の特徴（痕跡）を重点的に監視することで侵入が検出可能になるというものである．これにより侵入解析に必要な情報量を削減できた．本論文では，この痕跡を用いた侵入検出手法へ正規手続きデータベース（以降，正規手続き DB と表記）を用いた侵入判定手法を適用することを提案している．具体的には痕跡が検出された際にそれを引き起こしたプロセスについて，それが正規に認められた行為かということを正規手続き DB と照らし合わせ，認められた行為でなかった場合，侵入と判定するというものである．これにより未知の手法による侵入の検出が可能になる．

本論文の構成は，2章では既存の IDS の手法を分類し，各手法による長所や短所について述べる．また，既存の侵入検出システムの例をあげて説明する．3章では痕跡を用いた侵入検出手法および正規手続き DB の作成方法と，正規手続き DB を用いた侵入の判定方法について述べる．4章では，試作システムの概要について述べる．5章では，提案手法について考察する．

2. 既存の侵入検出技術について

2.1 用語定義

本論文でこれから用いるいくつかの言葉を定義しておく．

データソース 侵入検出に用いるデータのことで，主にシステムログとネットワークパケットを指す．

ローカルユーザ 対象ホストへのログイン権限を持つユーザ．

リモートユーザ 対象ホストへのログイン権限を持たないユーザ．

リモートアタック リモートユーザが，ネットワークを介して対象ホストへの侵入を試みること．ネットワークサービスの弱点やバグをついて侵入を試みることがこれにあたる．

ローカルアタック ローカルユーザが，権限を超えたアクセスを試みること．あるホストにログインした一般ユーザがそこで不正なコマンドを実行して管理者権限を得ることなどがこれにあたる．

2.2 既存の検出手法

これまで侵入検出に関して様々な手法が研究されてきた．従来の手法は分析するデータソースの種類と分析方法によって分類することができる．分析するデータソースにはホストの監査ログとネットワークパケットがあり，前者を用いて分析を行う IDS はホストベース IDS と呼ばれ，後者を用いるものはネットワークベース IDS と呼ばれる[6]．

データソースの分析方法で以下の2種類に分類される．

- AID（Anomaly Intruision Detection）
- MID（Misuse Intrusion Detection）

AID と MID の詳細は次に述べる．

2.2.1 AID

AID は「侵入行為が行われているとき，システムやユーザは通常の状態とは異なった振舞いをする」という仮定に基づいて，監査データを解析することにより侵入を検出する手法である．この手法には統計的手法[7]やニューラルネットワーク[8]などが用いられるが，ここでは AID において広く用いられている統計的手法について説明する[9]．

統計的手法では，ある時間間隔における様々な尺度（measure）を用いて，個々のユーザや対象のホストについての「通常の状態」を表すプロファイルを作成する．ここで尺度というのは，数値や2値で表現可能なユーザの行動やシステムの状態のことを指す．そして，そのプロファイルと現在の状態とを比較して，ある閾値を超えているものを侵入として検出する．尺度としては，ユーザについては平均ログイン時間や，システムリソースの利用量，特定のコマンドの実行回数や特定のファイルへのアクセス回数などを用いることができる．システムについては時間あたりの CPU 稼働率などを用いることができる．作成されたプロファイルは定期的に過去のものを統合していくことで，ユーザやシステムの長期的な変化にも対応させることが可能である．

AID を用いることの利点としては，

- 侵入を検出するのに侵入に関する知識が必要ないため，未知の侵入を検出できる可能性がある，

ということがあげられる．欠点としては，

- 生成されるログに対して統計処理を行うため，システムへのオーバヘッドが大きい，
- ユーザの行動をプロファイルに反映させる機能を侵入者が悪用し，徐々に侵入行動を通常の行動として反映させてしまう可能性がある，
- 侵入と通常行為を判定するための閾値の設定が難

しい，
- 通常行為と似通っているか，異なっているか，という観点でしか判断をしないので，侵入が検出された際に侵入に用いられたセキュリティホールが発見しにくい，

ということがあげられる．

AID の性能を決定づける主な要因は以下の 2 点である．

最適な尺度の組合せ どのようなデータでも尺度として用いることが可能であるが，侵入検出に有意な尺度群を論理的に決定するのは困難である．

閾値の設定 閾値が低すぎると，侵入でない行為を侵入としてしまうこと（フォルスポジティブ）が多くなり，逆に閾値が高すぎると，侵入を見逃してしまうこと（フォルスネガティブ）が多くなる．

2.2.2 MID

MID は，既知の侵入行為に関するデータをなんらかの形で保持しておき，収集したログとの比較により侵入を検出する手法である．たとえば，phf という CGI スクリプトを悪用して不正にコマンドを実行させるという攻撃[10]は，http サーバのアクセスログの中に .../cgi-bin/phf... という文字列を含むログがあるかどうかを調査することで検出することができる．これは，この攻撃を検出するためにはそういった調査を行えばよいという知識に基づく検出方法なので MID である．

MID を用いることの利点は，
- 侵入検出によるシステムへの負荷が少ない，
- 侵入者が，どのセキュリティホールを悪用して侵入したか分かる，

といったことがあげられる．欠点としては
- 登録されていない侵入行為は検出できないため，未知の侵入は検出できない，
- 新たに発見された侵入手法は，そのデータを追加しなければ検出できない，

といったことがあげられる．

MID の手法を用いた IDS を構築する際に重要になるのは，侵入行為をどのようにモデル化するかということである．侵入のモデル化にはエキスパートシステム[11),12)]や，状態遷移モデル[13)]，モデルベース推論[14)]などが用いられている．

2.3 既存の侵入検出システム

ここで前節で述べた手法を用いた既存の IDS として SRI☆で開発された NIDES を例にとり，そのシステムの概要を示す[15)]．

NIDES（Next-Generation Intrusion Detection Expert System）の目的は，リアルタイムにセキュリティ違反を検出するための，検出対象マシンの環境に依存しないシステムを提供することである．NIDES では AID と MID の両方の手法を組み合わせて用いることで検出可能な侵入手法の幅を広げている．NIDES は各ユーザの行動パターンを統計的なプロファイルとして保存しておきそのデータを現在のセッションと比較することで，通常とは異なった行動をしている疑わしいセッションを検出する．また，過去の侵入事例をもとに構築したエキスパートシステムにより，侵入の疑いのあるセッションを検出する．

NIDES は AID の手法として統計的手法を用いている．個々のユーザを特徴づけるプロファイルを 40 個以上の尺度を用いて作成する．これらの尺度は以下の 3 種類に分類される．

数値や量に関する尺度 アクセスしたファイルの数，更新されたディレクトリの数，など

2 値で表される尺度 特定のディレクトリにアクセスしたか，パスワードを変更したか，など

回数に関する尺度 コンパイラを使用した回数，特定のファイルにアクセスした回数，など

これらのプロファイルは定期的に更新されることで，個々のユーザの行動を反映してゆく．また，個々のユーザ以外にも個々のホスト，いくつかのホストから成るネットワーク，ユーザグループといった主体に対してもプロファイルが作成可能である．

NIDES はまた，MID の手法としてエキスパートシステムを用いている．既知の侵入手法や，侵入者に攻撃されうる監視対象ホストのセキュリティホールや，一般的に侵入と判断される行動パターンをルールベース化して，現在のセッションと比較し，マッチした場合これを侵入として検出する．新たに発見された侵入手法はセキュリティパッチなどにより，ルールベースに反映される．また，セキュリティホールの多くはホストの運用環境に依存するため，ルールベースは環境に合わせてカスタマイズが可能になっている．

3. 提案手法の概略

従来の侵入検出手法において，AID は未知の侵入手法による攻撃を検出できるが，正しいプロファイルの作成，維持という点に問題があり，MID は既知の侵入手法による攻撃はルールの作成により容易に検出できるが，未知の侵入手法による攻撃を検出できないという問題点があった．そこで NIDES のように両方の

☆ http://www.sri.com/

手法を用いることでこれらの問題点を互いに補い合うタイプの IDS も研究されたが，システムへ負荷がかかるという問題が生じた．その問題点を解決するために我々は，痕跡と正規手続き DB を用いた IDS を構築した．

提案手法は，ホストベースの IDS において痕跡とそれを引き起こした手続きという観点からデータソースを分析し，その手続きが正規のものであるかによって侵入を判断する．本章では痕跡という概念とそれに基づいた侵入検出手法について説明し，次に正規手続き DB とこれを用いた侵入の判定方法について説明する．

3.1 痕跡に基づく侵入検出手法

我々は，これまでに痕跡という概念に基づいた侵入検出手法を提案し[16]，侵入検出システム IDA（Intrusion Detection Agent System）を UNIX 上に実装した．これは，侵入の結果残される事象（痕跡）に注目することで効率良く侵入を検出する手法である．通常時にはログデータの蓄積と，データの中に痕跡が含まれるかの検査のみを行い，痕跡が発見されたときに，蓄積されたログデータから必要な情報を収集，分析してその痕跡が侵入の結果残されたものなのかどうかを判定するというものである．

我々が事前に行った侵入事例の調査から，その主な事例であるバッファオーバフロー攻撃やファイル変更攻撃にはある程度決まったパターンがあることが分かった．そこでまず，痕跡として一般ユーザによる管理者権限でのコマンドの実行と，一般ユーザによるセキュリティ上重要なファイルへの変更を定義した．そして収集したログとパターンとを比較することで痕跡が侵入の結果残されたものかどうかを判定した．この手法は，過去の侵入事例から攻撃パターンを見つけ侵入判定に用いるという点では，MID に分類することができる．

この手法を用いることでシステムへのオーバヘッドを軽減させるという利点が得られた．また個々の攻撃からパターンを抽出したことにより，このパターンにあてはまる侵入手法であれば，未知の攻撃であっても検出することができるという利点も得られた．たとえば，従来の IDA ではバッファオーバフロー攻撃の検出が可能であるがバグのある個々の特権プログラムについて検出のルールを決めているわけではないので，新たに特権プログラムにバグが見つかっても，それを悪用したバッファオーバフロー攻撃は検出が可能であるということである．しかし，パターンに合致しないような新しい手法による攻撃は，検出することができないという問題点が残った．

そこでこの問題点を解決するために，検出された痕跡に対する新しい判定方法として，痕跡が正規の手続きの結果残されたものかどうかを調べることで侵入かどうかを判断することにした．この手法を実現するためにはそれぞれの痕跡について，それを正規に残すことができる手続きにはどのようなものがあるかをデータベースとして保持しておく必要がある．そして痕跡を検出した際に，痕跡の原因となった行為とこのデータベースを照らし合わせ，データベースにない手続きによって残された痕跡を侵入と判断する．これにより未知の手法による侵入が検出可能になる．このデータベースを正規手続きデータベース（正規手続き DB）と呼ぶ．我々はこの提案手法を UNIX 上で実装した．次節で正規手続き DB の作成方法と，正規手続き DB を用いた侵入の判定について述べる．

3.2 正規手続き DB による侵入の判定

3.2.1 正規手続き DB の作成に必要な情報

正規手続き DB を作成するためには，どのような正規のユーザ行為によって痕跡が残されるかを知らなければならない．痕跡を一般ユーザによる管理者権限の取得と定義すると，一般ユーザが root に SUID されたコマンドを実行するときに痕跡が残される．たとえばパスワードを変更するには/etc/passwd, /etc/shadow などにアクセスする必要があるので，passwd や admintool☆といった SUID コマンドを実行して変更を行うが，このとき，一般ユーザによる管理者所有の重要ファイルへの変更という痕跡が残るのである．したがって，正規手続き DB は SUID コマンドを調査することで作成することができる．root に SUID されているコマンドには以下の 3 種類がある．

(1) 特定の root 所有ファイルにアクセスする（passwd，admintool）．
(2) プロセスの属性を変更する（su，login）．
(3) 特権ポートを使用する（rcp，rlogin）．

痕跡を root 所有のファイルへのアクセスとした場合，(1) に含まれるコマンドにより root 所有のファイルへのアクセスが可能になるため，(1) の SUID コマンドに許可されているセキュリティ上重要なファイルへのアクセス権限をあらかじめ記述しておけば，ファイル変更の痕跡が発見された際に，それが許可されているかを調べることが可能になる．**図 1** に正規手続き DB の記述例を示す．

ファイル内容，モード変更の痕跡が検出されたとき

☆ ファイル名やコマンド名は SUN Microsystems Solaris2.7 のケースである．

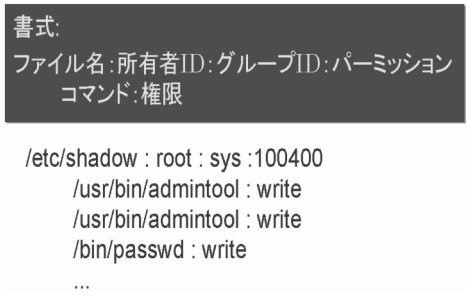


図 1 正規手続き DB の例
Fig. 1 Example of legitimate access data base.

に，痕跡を引き起こしたコマンドが正規手続き DB に登録されていて，かつ対象となるファイルへのアクセスを許可されていた場合には，その痕跡は侵入ではないと判断される．

3.2.2 正規手続き DB の新規作成

ホストにインストールされている SUID コマンドの数や種類はそのホストの環境に依存する．したがって，正規手続き DB は，ホスト固有のものを用意しなければならない．ここでは監査対象のホストに提案手法を用いた IDS を導入する際のそのホスト固有の DB を定義するための方法について述べる．具体的には以下のような手順により SUID コマンドの登録を行う．

(1) IDS 導入時に含まれている SUID コマンドを `find` ですべて抽出する．
(2) 各コマンドについて付属のマニュアルにより調査する．

コマンド付属のマニュアルにより，利用方法やアクセスするファイルの情報を得られる．もしそこに記載されていないような方法で利用されたときは，そのコマンドの意図した利用方法から逸脱しているとして，侵入と判断することができる．したがって，マニュアルに従い正規手続きを定義すれば，コマンドのバグを利用した侵入を検出できる可能性がある．

4. 試作システムの実装と評価

我々は提案手法の実現可能性と実データに基づいた実用性の評価を与えるために試作システムを構築した．本章ではこの試作システムについて述べる．

4.1 痕跡の定義

提案手法を実装するために，インターネット上で取得したクラックツールのうちローカルアタックに分類されるもの☆158 件について分析した（**表 1**）．その結果不正な root 権限のプロセスを起動する攻撃が 111 件，任意のファイルに変更を加える攻撃が 36 件，任意のファイルモードを変更する攻撃が 5 件，それ以外の攻撃が 6 件であった．そこで調査した範囲内で多数を占めていた不正な root 権限のプロセス起動と，任意のファイルへの変更，任意のファイルのモードを変更する攻撃を検出対象とし，痕跡を以下のように定義した．この定義は従来の IDA における痕跡の定義と同じである．

ここでセキュリティ上重要なファイルとは，

- /etc/passwd
- /etc/shadow
- /etc/hosts.equiv

などである．root 所有のファイルはいくつかあるが，侵入が行われた際に改竄されるのは主にこれらのファイルである．

☆ サービス不能攻撃は除く．

表 1 攻撃の種類と痕跡
Table 1 Exploit and MLSI.

攻撃の種類	痕跡
一般ユーザの root 権限でのコマンドの実行	実効 UID が root の SUID されていない exec()，execve() システムコールの実行
root 所有のセキュリティ上重要なファイルへの書き込み	open()，create()，rename() システムコールの実行
root 所有のセキュリティ上重要なファイルの属性変更	chmod()，chown() システムコールの実行

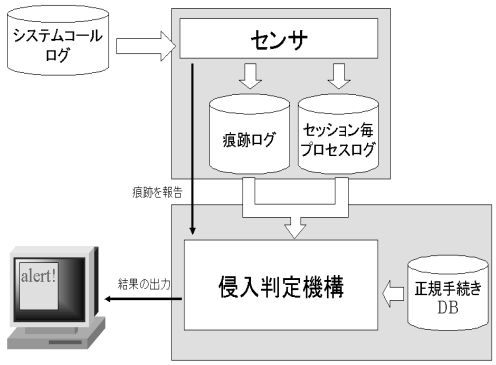


図 2 試作システムの全体図
Fig. 2 Prototype system.

4.2 システム構成

提案するシステムは以下のコンポーネントから構成される（**図 2**）．

- センサ
  対象ホストで収集されたログを各セッションごとのログに分類して保存する．そして，痕跡となるデータが検出されたときに侵入判定機構へ報告を行う．

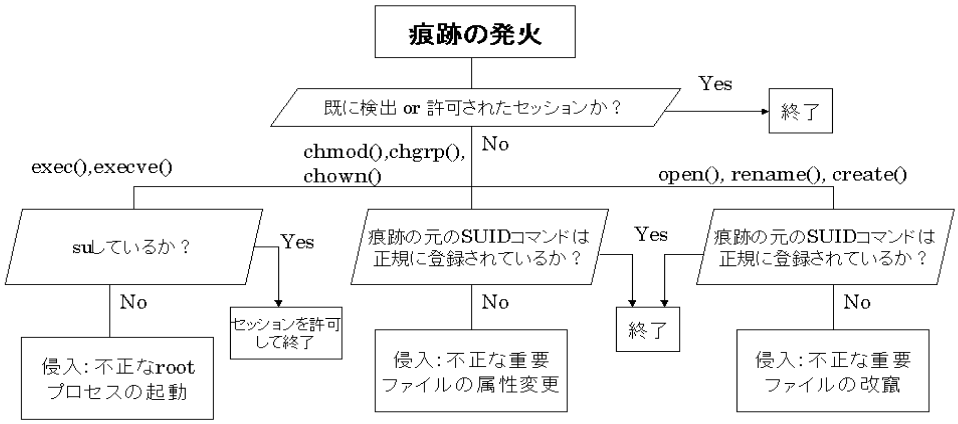


図 3 痕跡の検出から侵入の判断
Fig. 3 Judging MLSI.

- 侵入判定機構
  センサから痕跡が検出されたという報告を受けると，その痕跡に関する情報をセッションごとのログから抽出し，正規手続き DB を参照してそれが侵入かどうかを判断する．
- 正規手続き DB
  正規の手続きを痕跡の種類ごとに記述したもの．
- セッションごとプロセスログ
  収集されたログをセッションごとにまとめたもの．

4.3 痕跡の発見から侵入検出までの流れ

提案手法による侵入検出の流れを以下に記述する．

(1) 対象セッションがそれ以前に侵入と判断されているか確認．
(2) 痕跡の種類を判断．
(3) 痕跡の元のプロセスのコマンドを調査．
(4) 元プロセスのコマンドが正規手続き DB に登録されているかを確認．
(5) 登録が確認されれば終了する．このとき，痕跡の種類が exec()，execve() のときはそのセッションを許可して終了する．
(6) 登録されていなければ侵入を警告する．

処理の流れを**図 3** に示す．

4.4 開 発 環 境

我々は，試作システムを以下のような環境で開発した．実装量はセンサ部（Perl）が 402 ステップ（8 KB）で，侵入判定機構（C 言語）が 523 ステップ（9 KB）である（**表 2** 参照）．

4.5 試作システムでの実験

試作システムの侵入検出能力とパフォーマンスを評価するために実験を行った．

表 2 開発環境
Table 2 Building environment.

CPU	Sparc，AMD K6-2 300 MHz
OS	Solaris2.7，Vine Linux1.1（Kernel2.0.36）
言語	gcc 2.7.2.3，perl 5.004_04
ログツール	Basic Security Module [17]（Solaris）， 独自に開発したログツール（linux）

4.5.1 クラックツールによる実験

試作システムの検出能力を評価するために，インターネットから入手したクラックツールを用いて侵入をシミュレートし，試作システムが侵入を検出できるかどうかの実験を行った．その結果先に述べた 158 件のクラックツールにおいて，Solaris と linux を攻撃対象としたもので，動作確認ができた 13 件のツールのうち 12 件は検出可能であった．ここで検出されなかったのは，スニッフィングのような，痕跡を定義することができない攻撃であった．

4.5.2 実運用環境での実験

試作システムのパフォーマンスを評価するために，実運用環境を想定して一般的なクライアントホストで 8 名のユーザ（うち，管理者権限を持つ者 2 名）による 10 日間の実証実験を行った．その結果**表 3** のようなデータが得られた．IDS がシステムに与えるオーバヘッドの量はそのシステムの置かれた環境や利用状況をに大きく左右される．したがって今回の実験のみで様々な環境下でのシステムオーバヘッドに対する評価を下すことはできない．しかし，得られた結果を 1 つの参考とすることはできる．なお，今回の実験では，正規手続き DB の更新作業は発生しなかった．

表 3 実験結果
Table 3 Result.

総セッション数	75 セッション
痕跡と判定されたイベント	30 イベント
侵入と判定されたイベント	0 イベント
痕跡ではないシステムコールの平均処理時間（CPU 時間）	0.03 秒
痕跡となるシステムコールの平均処理時間（CPU 時間）	0.05 秒
蓄積されたログの量（75 セッション分）	215 KB

5. 考　　察

これまで，痕跡と正規手続き DB を用いた侵入検出手法の概要について述べてきた．ここでは，この手法に関するいくつかの議論について述べる．

正規手続き DB の作成方法について　今回，我々は正規手続き DB を SUID コマンドの網羅的な調査によって作成した．コマンドの網羅的な調査のほかにデータベースを作成する方法には，実運用環境下でそれらのコマンドに関するデータを収集し，得られたデータからデータベースを作成するという方法が考えられる[18),19)]．この方法ではデータベースの作成期間を IDS の運用前に設ける．

コマンドを網羅的に調査する方法は比較的定義洩れの心配がないという長所がある．しかし，このような調査，登録作業は繁雑であり，また，マニュアルなどの情報がないために調査が困難な場合がある．一方，実運用環境下でデータを収集する手法は，実際にコマンドを利用したときのデータを用いるためデータの取得が楽である反面，必ずしも網羅的なデータではないので，そこから作成されるデータベースは不完全なものになる可能性がある．そのため，本当は正規の行為であるにもかかわらずデータベース作成段階で現れなかったために侵入と判断されるケース（フォルスポジティブ）があるという欠点がある．我々はこのフォルスポジティブを避けるため，前者の手法を採用した．

提案手法の有効性，実用性　正規の手続きでない行為を侵入として判断するので，具体的な侵入手法がどのようなものであるにせよ，痕跡の事象を含む場合にはすべて検出が可能である．それゆえ，同一痕跡を残す未知の侵入も検出が可能になる．したがって，提案手法は痕跡と正規手続き DB を利用した侵入検出によって未知の侵入手法が検出可能である．また実運用実験の結果より，この試作システムがホストにほとんど負荷をかけていないということが分かった．

提案手法の適用範囲　この提案手法はシステムコールレベルのログがとれるような UNIX オペレーティングシステム上で実現可能である．我々は試作システムを Solaris，Linux 上で実装した．また我々は，ローカルアタックを侵入検出の対象とした．なぜなら，ローカルアタックは攻撃対象ホストでの管理者権限の取得が主な目的であることが多く，リモートアタックに比べて危険度がより高いと思われたからである．なおスニッフィングなど痕跡が定義できない攻撃は，現在の検出機構では検出できない．また DOS 攻撃はリモートアタックなので現在は検出対象外である．

提案手法と従来技術との関係　痕跡を用いた手法は，従来技術と比較した場合，AID，MID といった観点で異なるのではなく，データの処理方法の観点で異なる．従来技術（AID，MID を問わず）では出力されるログを随時分析するデータドリブン型であるのに対し，痕跡を用いた手法では，痕跡が残されたときにログの分析を行うゴールドリブン型である．正規手続き DB を用いた手法はユーザ行動の正規性より判断しているという意味で AID の一種であるが，従来の AID のようにすべての行動をモニタし統計処理する必要がないため，システムに負荷がかからない．

従来の IDA との比較　提案手法では正規手続き DB を利用することで従来の IDA では検出できなかった，未知の侵入手法による攻撃が検出可能になる．また既知の侵入手法による攻撃に関しては従来の IDA と同等の検出精度であった．未知の侵入手法に関しては，現在実運用環境下での実験を十分に行っていないため定量評価は困難であるが，今後実運用環境下での実験を十分に行うことで，評価していくことができるものと思われる．

6. ま と め

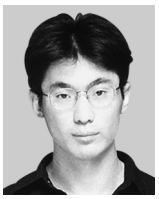
痕跡を用いた侵入検出手法に正規手続き DB を利用した侵入の判定を適用することで，侵入手法に関する知識を必要としない侵入検出が可能になった．またシステムに負荷をかけずに未知の侵入手法による侵入が検出可能になった．今後の課題としては正規手続き DB の自動作成とリモートアタックへの対応があげられる．

謝辞　本研究を進めるにあたって，ご指導いただいた永田守男教授，実装面に関するアドバイスをしてくださった IDA 開発メンバーの皆様に深謝いたします．

参 考 文 献

1) http://www.cert.org/
2) http://www.ciac.org/
3) Lunt, T.: A surbey of intrusion detection techniques, *Computers & Security*, 12, pp.405–418 (1993).
4) Kumar, S. and Spafford, E.: An Application of Pattern Matching in Intrusion Detection, Technical Report, 94-013, Purdue University (1994).
5) Asaka, M., Taguchi, A. and Goto, S.: The Implementation of IDA: An Intrusion Detection Agent System, *Proc. 11th FIRST '99* (1999).
6) Mukherjee, B., Heberlein, L. and Levitt, K.: Network Intrusion Detection, *IEEE Network*, May/June, pp.26–41 (1994).
7) Javitz, H., et al.: The NIDES Statistical Component Description and Justification, Annual Report, March 7 (1994).
8) Lunt, T.: Detecting Intruders in Computer Systems, *Conference on Auditing and Computer Technology* (1993).
9) Denning, D.: An Intrusion Detection Model, *IEEE Trans. Softw. Eng.*, Vol.SE-13, No.2, pp.222–232 (1987).
10) http://www.jpcert.or.jp/ESA/CA-96.06.euc.txt
11) Lunt, T., et al.: Real Time Intrusion Detection, *Computer Security Journal*, Vol.6, No.1, pp.9–14 (1989).
12) Dowell, C. and Ramstedt, P.: The COMPUTERWATCH data reduction tool, *13th National Computer Security Conference*, pp.99–108 (Oct. 1990).
13) Ilgun, K.: State Transition Analysis: A Rule-Based Intrusion Detection Approach, *IEEE Trans. Softw. Eng.*, Vol.21, No.3, pp.181–199 (1995).
14) Garvey, T. and Lunt, T.: Model Based Intrusion Detection, *Proc. 14th National Computer Security Conference*, pp.372–385 (1991).
15) Anderson, D. and Lunt, T.: *SAFEGUARD FINAL REPORT: Detecting Unusual Program Behavior Using The NIDES Statistical Component* (1993).
16) Asaka, M., et al.: Local Attack Detection and Intrusion Route Tracing, *IEICE Trans.*, Vol.E82-b, No.11, pp.1826–1833 (1999).
17) SunSHILD 基本セキュリティモジュール, Sun Microsystems, Inc (1995).
18) Forrest, S., Hofmeyr, S. and Somayaji, A.: Computer Immunology, *Comm. ACM*, Vol.40, No.10, pp.88–96 (1997).
19) Forrest, S., Hofmeyr, S. and Somayaji, A.: A Sense of Self for Unix Processes, *Proc. 1996 IEEE Symposium on Security and Privacy*, Los Alamos, CA, pp.120–128 (1996).

(平成 11 年 11 月 30 日受付)
(平成 12 年 6 月 1 日採録)



原田 慎介

昭和 50 年生. 平成 10 年慶應義塾大学理工学研究科管理工学専攻修士課程入学. 侵入検出システムの研究に従事.

浅香 緑（正会員）

昭和 59 年上智大学大学院博士前期課程数学専攻修了. 株式会社日本総合研究所勤務. 平成 5 年より情報処理振興事業協会に出向中. 侵入検出システムの研究に従事. 電子情報通信学会会員.